HeartCare: an Agent Oriented Architecture Implemented with Actors*

Mariana Falco[1][0002-0959-7435], Leandro Bauret[2], and Gabriela Robiolo[3]

[1] LIDTUA/CONICET, Engineering Faculty, Austral University
mfalco@austral.edu.ar
[2] Engineering Faculty, Austral University
leandro.bauret@ing.austral.edu.ar
[3] LIDTUA, Engineering Faculty, Austral University
grobiolo@austral.edu.ar

Abstract. A priority in cardiac patients is the performance associated to a regular follow-up, when they leave the hospital to keep track on their scheduled physical activities. Unfortunately, not all patients begin a rehabilitation program, or if they are part, they do not perform all of the exercises. Consequently, a multi-agent system was designed and developed to enable patients with heart conditions and their doctors to achieve a feasible out-of-hospital follow-up; while monitoring their heart-rate measures and physical workout. The present paper describes how the architecture was built in which Akka and Scala have been used to represent the agents through the actors model. We can conclude that it was a good decision to have used the agent-oriented paradigm for the future scalability of the application (both for the increase in the number of sensors and the possibility of agents with beliefs), but not for the CRUD operations. Future work will include a new development of Heart Care, adding more sensors and newer layers of intelligence.

Keywords: Software agents · Actors · Monitoring of patients.

1 Introduction

Between 2005 and 2015 in Argentina, the number of deaths from heart attacks was increased by 18 percent. It is estimated that between 40,000 and 50,000 strokes occur each year; and of that total, 17,000 people die. The main causes are the delay in the attention or intervention of a doctor in time due to the large number of patients who appear in hospitals [17] and in-hospital mortality. Due to a problem of organization and the lack of adequate technology for more efficient assistance to a larger number of people, the number of deaths due to cardiac disorders is increasing [1].

* The authors thank Tomás Fernández Martínez, Ignacio Berdiñas, Nicolás Velázquez Bustamente, Matías Cicilia, Ignacio Vázquez, and Agustín Masuelli, who collaborated in the development and design of HeartCare. This work is supported by a Research Grant from Austral University.

A major priority in cardiac patients is the performance related to a regular follow-up, while improving the level of medical care [13]. Consequently, an important goal for achieving an improvement with a cardiac patients physical fitness is for them to realize regular physical activities [2, 3], because one of the most important risk factors is physical inactivity [4, 5]. Some patients do not access a cardiac rehabilitation program, do not complete their weekly physical routines, or do not workout at all [6, 7]; for several reasons like lack of transportation, the patient is unaware - he or she does not have a confirmation of the follow-up, lack of support, economic issues, among others.

As technology progresses, its uses entail the creation of various paradigms such as u-healthcare, pervasive healthcare, and others, which allow to move the practice of doctor's office to the patient's daily life [8, 9], through mobile devices, various sensors, smart-glasses, among others. Specifically in cardiac patients, there are various web / mobile applications [10–12] that allow them to control their symptoms, while achieving adherence to their treatment through the management of medications, physiological values, possible symptoms and how to act in front of them, physical exercise routines, and even by reading healthy tips.

There are different studies that approaches this problem from several disciplines, through for example, a system for out-of-hospital follow-up of cardiac patients that belong to one of four risk groups (arterial hypertension, malignant arrhythmias, heart failure, and post-infarction rehabilitation) [13], a multi-agent architecture for heart failure management in a home care environment [18], and an ontology-based framework to support cardiac rehabilitation program [14].

Within Artificial Intelligence, multi-agent systems [21] helps with the development of complex systems, and healthcare is a complex environment. Patient management has seen the most applicability of agents in healthcare [23], although nowadays the whole spectrum is covered. Isern and Moreno [15] define seven types of agent-based applications based on different domains: data-management systems, secure platforms, decision support systems, planning, simulation, care platforms, and monitoring and alarms. The utilization of sensors and wearable devices which monitors the actual state of the patients helps improving the living-at-home systems or ambulatory activities, in order to assist the patients on improving their condition [15].

The present paper tries to answer the following research questions: (RQ1) Is the multi-agent paradigm suited for the application for cardiac rehabilitation?; and (RQ2) What is the most appropriate framework to build the application?. In this context, we have developed an agent-based application called HeartCare that allows the monitoring and medical care of cardiac patients, in order to support ubiquitous contact between chronic patients presenting cardiovascular risk and their doctors or therapists, directly involved in their out-of-hospital follow-up.

The possibility of registering several parameters according to the needs of the patients or the pathology that determines the clinical condition of the patient in ambulatory or domiciliary form is a novelty. It will allow to see possible changes in the evolution of the disease, either improvements with medication or deterioration in the absence of adequate treatment / diagnosis, and over time to establish predictive values of certain parameters according to the evolution of the disease by conducting studies.

Our main goal here is to present the architecture of the HeartCare application while discussing the decisions made with respect to the agent-oriented paradigm, the technologies, and its implementation. Also, specifying the agents developed, and each of their goals and behaviors. HeartCare is a prototype version of the architecture, and we have implemented the actor's model to represent the agents through Akka [28].

The work is structured as follows: section 2 will describe the application, the medical context addressed, the functionalities implemented and the technologies used. Section 3 will address the definition of the architecture, specifying the agents implemented, and the cooperation between them by means of a sequence diagram. Section 4 will describe the conclusions and future work.

2 HeartCare: Scope and Technologies

2.1 Understanding the scope

The HeartCare application aims to assist in home-based rehabilitation of patients who have suffered cardiovascular accidents, specifically in out-of-the-hospital follow-up. The main objective is to ensure that the recovery can take place in an environment outside hospitals, without the need to attend monthly appointments with a doctor at hospitals with the only purpose of doing a follow-up of the patient's health status and establish the next steps to follow.

Within the medical context and particularly in cardiology, there are two main classifications that place patients on a category based on their limitations during physical activity [11]. The first one, a three phase classification that defines the treatment for patients who have suffered some cardiac affection, and the second one, four classes derived from the NYHA functional classification. In this context, the HeartCare application will seek to accompany patients in phase II (active phase or supervised exercises) and phase III (maintenance phase), and also class I patients that are able to perform physical activity autonomously without affecting their health.

According to the application's goal, it is possible to determine two main users: patient and health staff, where the latter are responsible for monitoring the clinical history, and health condition of the patient. Being HeartCare a mobile and web application, it can be used on computers, mobile devices or tablets. Both patients and specialists require an internet connection in order to use the application. HeartCare provides a mobile patient-oriented application to assist patients in their cardiac recovery.

For the health specialists, HeartCare has a staff-oriented web version to track the health condition and status of their patients. The defined use cases are grouped by functionality with the following criteria:

- `User management`: It allows the patient and the doctor to create a user, and also to login and logout.

- `Measure management`: It allows the patient to add new measurements on the mobile application, and to the doctor as well on the web version.
- `Routine management`: It allows the patient to visualize the set of routines he/she has to do, visualize each exercise, interact with the routine (play, pause, stop), and fill the Borg Scale (explained below).
- `Specialist management`: It allows the doctor to visualize patients and their data, to see measurements registries, and interact with trends graphs.

The patient through his or her smart-phone can visualize the routine defined by its doctor. Each routine contains a set of exercises (like warm-up, stretching, walking, among others) on a specific order, the amount of series or repetitions for each exercise, and the expected Borg Scale - set by the Doctor. Once the patient finishes an exercise (all of the series within each exercise), the application will ask the user to rate the level of perceived exertion according to the Borg Scale.

This scale relatively matches how the patient is feeling while working out, through a number from 6 (none) to 20 (extremely hard). The selection of the numbers (6 to 20) by its creator Dr. Gunnar, was due to a simple way to estimate heart rate by multiplying the number or score selected by 10 which gives an approximate heart rate value for a particular level of activity [24].

Based on the score selected for each exercise, when the level is hard, the patient is having a symptom like dizziness, the heart rate levels are not regular for a person performing physical exercise, the Agent Patient considers that the patient is at risk and so, an alarm to the doctor is sent for him to analyze the severity, and change the plan of exercises or ended due to complications.

2.2 Technologies used

Once the scope of the application was defined, it was necessary to decide which technologies were right for the requirements of the application (web version, mobile version, connection to a heart rate sensor through Bluetooth, monitoring in real time, increased concurrency with more sensors). After analyzing agent-oriented technologies, reading documentations and following some test cases, the team decided that the agent-oriented technologies, most of them described by Agent-Oriented Software Engineering (AOSE), were not a perfect fit for Heart-Care. So, the main difference with most of the developed agent-oriented applications [22] is that we did not choose JADE [27].

Concurrent systems are composed of a large number of computations that run simultaneously and can interact with each other [25]. The execution of computations in concurrent systems can be done in different ways, therefore, several models have emerged that allow the modeling of concurrent systems. One of them is the model of actors, adopted by Scala; in which actors can be seen as the replacement of threads in traditional multi-threaded applications, which are able to achieve interaction with other actors through the concepts of message passing. A key point within this model is that an actor can react to a message received in three different ways: a) creating a finite number of new actors, b) sending a finite number of messages, and c) controlling its internal state by updating its mutable attributes [30]. HeartCare is expected to have a high number of sensors transmitting at the same time.

Agent-based systems require frameworks with high performance, and concurrent simulation, therefore, it is possible to represent agents through actors. The authors in [26] have implemented an evolutionary multi-agent system (EMAS), while introducing the concept of meeting arenas which allows a more scalable implementation of MAS. A meeting arena is defined by [26] as: Every time an agent wants to perform an action, it chooses an appropriate arena to meet with other similar agents. In their work, the agents and the arenas have been implemented in Akka [28], where the actors execute asynchronously and communicate through message passing.

Based on the above, the second main difference is that we selected an actor-oriented framework (Akka) that allowed us to build the agents by implementing actors and classes. We modeled a multi-agent system with different intelligent agents, with the particular concepts of agency (like flexible behavior, beliefs, desires, and intentions, communication through ACL messages, among others), but as a point of implementation we have coded actors with a layer of intelligence in order to achieve these concepts. One important characteristic is the emergent behavior that arose as a result of the evolution of the patients, evidenced by vital signs and medical data records, which we hope will be a consequence of the analysis of the recorded data.

As such, and in particular for the web version, TypeScript was used as the main language, making use of a web development framework called Angular combined with a web interface framework called UiKit , and a useful library for the creation of graphics named Angular2 Charts. For the mobile version, Javascript was used as the main language, using a native application development framework called React Native. The back-end consists of Scala as the programming language, Akka as a development framework building an agent model with the actor model, Slick as the ORM (Object-Relationship Mapping), and PostgreSQL as a database system. Finally, the sensor used for the implementation of HeartCare is a Polar H10 heart rate sensor [29] that allowed us to control the heart rate with high precision and connect it simultaneously to Bluetooth devices.

3 Describing the architecture of HeartCare

The architecture of the application can be seen in `figure 1`, which is composed of three modules: the mobile client, the web client, and the server consisting of the multi-agent system and the database. The mobile client is responsible for serving the patient's data through a mobile application so that he or she can perform the routines corresponding to his or her recovery. Patient data is collected using the Polar H10 heart rate sensor [29], and sent from the mobile client to the system. On the other hand, the web client provides health specialists with a user interface that allows them to view information on the status and progress of their patients. Finally, the server is comprised of the multi-agent system that is responsible for making decisions and queries of information to the database.

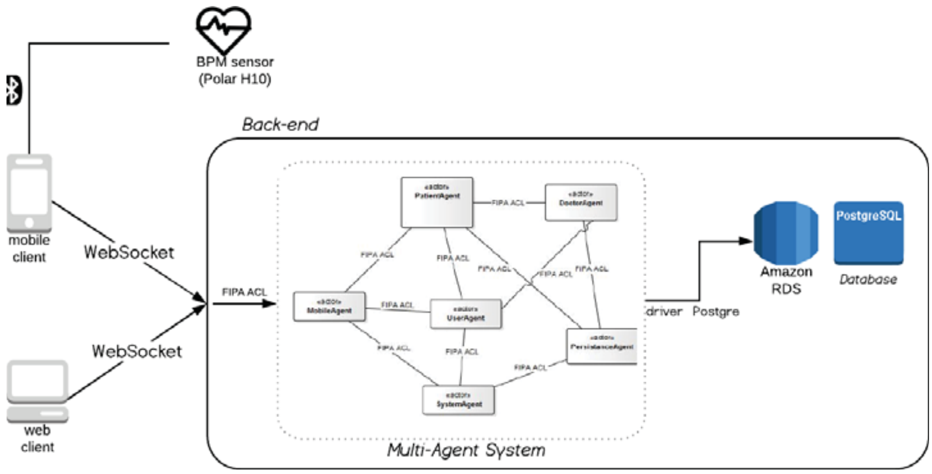

Fig. 1. The HeartCare architecture is displayed by showing each of the modules.

3.1 Specification of the agents

Ferber described that a multi-agent system '*(...) contains an environment, objects and agents (the agents being the only ones to act), relations between all the entities, a set of operations that can be performed by the entities and the changes of the universe in time and due to these actions (...)* [31]. In our context, the built MAS is composed of the following agents: MobileAgent, PatientAgent, HealthAgent, PersistanceAgent (modelled as a generalization relation with the PersistanceAgent as the superclass, and UserPersistanceAgent, PatientPersistanceAgent, RoutinePersistanceAgent, and MeasurePersistanceAgent as the subclasses), and System Agent.

All of the agents possess a mixed design between the Touring Machine Architecture [19] and the Layered Horizontal Architecture [20]. The layers resemble the Touring architecture but differ in that only one determines the response to input, as in the horizontal layer architecture. The agents are specified as follows:

`Persistance agents`: These five agents are responsible for persisting the database, and carrying out the CRUD operations of entities. They are the ones which know most about the domain of the problem since they understand all the entities that compose it. They do not have a planning layer since they do not make decisions based on their internal state.

`MobileAgent`: The MobileAgent contains the reference to the mobile application and is responsible for answering all customer requests and sending the necessary data to be stored in the agent system. Almost all its behavior is reactive. It has two layers, one reactive and the other planning.

`SystemAgent`: The SystemAgent is in charge of controlling the creation of the rest of the agents in the system, and it contains the context of the application and the system of existing actors. It is responsible for managing the user authentication of the system, receiving requests for user creation and login. It has a reactivity layer much larger than that of planning, and it does not have a model layer since it does not know about the domain of the problem.

`UserAgent`: The UserAgent represents the operations on the logged in user. This agent modifies its behavior based on whether the user is a patient or a health agent type. In the case that the user is a patient, then the one who will respond to the messages will be the PatientAgent. Otherwise, if the user is a health agent, he will adopt the behavior of a DoctorAgent. This is because the functionalities and messages available to a patient are not the same as those of the doctor, since they cannot access the same data and have different objectives per se.

`DoctorAgent`: The structure of the DoctorAgent is similar to that of the PatientAgent. Even though its planning layer is not currently developed, it aims in the future to allow the doctor to assemble the routines based on the objectives of each patient. The main objective of this agent is to achieve a successful cardiac recovery for all his patients following their particular exercise planning and schedule. It is worth mentioning that its structure is equal to the Touring Machine. This agent retrieves the information of the patients of a certain specialist; and is responsible for receiving new observations that the doctor logs into the system for a particular patient.

`PatientAgent`: The PatientAgent is the one that has as an explicit objective of a proper cardiac rehabilitation of the patient it represents. This is why when starting an exercise, the agent will be responsible for continuously analyzing the patient's condition and the context in which it is located and will make the best possible decision in order to achieve the recovery of the user. This is done through a BDI architecture [16], where the agent retrieves the available data about the user in question, analyzes what were his intentions through the history of execution of patient routines and, in response, executes the best plan that fits the agent's wishes,a plan that is fit in the best way to the wishes of the agent since at all times the PatientAgent will thrive in order to accomplish its objective. In case the agent interprets that the patient is in danger, it will react in such way to prevent and mitigate the risk to which the user is exposed. Otherwise, if the patient is in normal conditions, then the agents behavior of the patient will be in accordance with what was planned to achieve his goal.

3.2 Lessons learned

Regarding the first research question (Is the multi-agent paradigm suited for the application for cardiac rehabilitation?), it is possible to say that the decision to implement the agent-oriented paradigm (AOP) for the HeartCare application was appropriate in the following topics: (a) scalability, in the future new sensors will be added to improve the intelligence as part of the analysis of the patients condition, and it will also increase the concurrency between them; (b) considering the concurrent use of the system by several users, AOP allows a separate and independent thread of control that represents the state of each user and has modeled as its only desire its specific recovery, seeking to achieve their design objectives based on the data collected about the patient. The biggest disadvantage was observed when implementing the CRUD operations, which require the communication of multiple agents to only perform an update in the database; consequently a simple task is complexed enormously and the process itself ends up being inefficient.

Numerous developments and applications have been made for MAS and although, JADE [27] was found as the most used framework [22], there is a great dispersion in terms of quantity of existing frameworks, the community tends to create new ones instead of unifying existing ones. In this context, and regarding the second research question (What is the most appropriate framework to build the application?), it is feasible to mention that before a rigorous study and analysis, HeartCare was not implemented with any of the existing agent-oriented framework, methodology or programming language. Instead, it was implemented with day-to-day technologies like Akka and Scala. JADE was not selected as the framework, due to its deprecated Android version, the deficiencies of the SOAP protocol, the lack of future maintainability, even though it is open source it does not have a public repository on Git, and it is a verbose framework. One of the main characteristics is that this technological decision allowed us to implement actors while constructing a MAS, by using Akka. We can conclude that it was a good decision to implement the project in Akka since it streamlined the development mainly due to its greater documentation and the large number of people using this framework nowadays.

4 Conclusions

Healthcare systems are complex and highly dynamic, they manage distributed data and resources, and they usually need to achieve interoperability with different information systems. Likewise, the healthcare professionals and staff need a high mobility within the hospital or health institution [23]. As such, several applications have been made to contribute to these topics [22, 15]. In the present paper, a mobile and web application called HeartCare has been presented which main goal is to assist in the home rehabilitation of patients who have suffered cardiovascular accidents. The architecture was built with Akka and Scala have been used to represent the agents through the actors model.

From the realization of the architecture, and as future work, the following proposals arise to improve the current implementation: (a) develop the BDI logic of the HealthAgent so that it can create general plans, and even specific plans taking into account the current state of health and the patient's medical history; (b) improve the planning layer of the PatientAgent so that it is more related to the execution of the patient's routine; (c) add several sensors to represent with a higher accuracy the state and the environment in which the patient is, improving the system's robustness; (d) train even more the intelligence of the agents for the determination of the routines and a more precise recognition of the danger thresholds; and (e) the use of machine learning in order to identify behavior patterns while monitoring the patients.

Consequently, HeartCare will be able to propose more or less demanding routines, increasing the level of difficulty as he considers it appropriate to achieve his main objective in the most effective way. Also, it is expected that a patient's agent will interact with that of another patient and provide a team routine, selecting a routine in common to meet the objectives of both users.

References

1. Clarin, (2017), En 10 años crecieron 18 las muertes por infartos y apuntan a las demoras en la atención. https://www.clarin.com/sociedad/10-anos-crecieron-18-muertes-infartos-apuntan-demoras-atencion_0_rkN5SyTU-.html Last access: 23/06/2019
2. Fletcher, B., Magyari, P., Prussak, K., and Churilla, J. (2012). Entrenamiento físico en pacientes con insuficiencia cardíaca. Revista Médica Clínica Las Condes, 23(6), 757-765.
3. Leon, A. S., Franklin, B. A., Costa, F., Balady, G. J., Berra, K. A., Stewart, K. J., ... and Lauer, M. S. (2005). Cardiac rehabilitation and secondary prevention of coronary heart disease: an American Heart Association scientific statement from the Council on Clinical Cardiology (Subcommittee on Exercise, Cardiac Rehabilitation, and Prevention) and the Council on Nutrition, Physical Activity, and Metabolism (Subcommittee on Physical Activity), in collaboration with the American association of Cardiovascular and Pulmonary Rehabilitation. Circulation, 111(3), 369-376.
4. Platter, M., Hofer, M., Hlzl, C., Huber, A., Renn, D., Webb, D., Hfer, S. (2016). Supporting cardiac patient physical activity: a brief health psychological intervention. Wiener Klinische Wochenschrift, 128(5-6), 175-181.
5. Doukky, R., Mangla, A., Ibrahim, Z., Poulin, M. F., Avery, E., Collado, F. M., ... and Powell, L. H. (2016). Impact of physical inactivity on mortality in patients with heart failure. The American journal of cardiology, 117(7), 1135-1143.
6. Redfern, J., Ellis, E. R., Briffa, T., and Freedman, S. B. (2007). High riskfactor level and low riskfactor knowledge in patients not accessing cardiac rehabilitation after acute coronary syndrome. Medical Journal of Australia, 186(1), 21-25.
7. Ferrier, S., Blanchard, C. M., Vallis, M., and Giacomantonio, N. (2011). Behavioural interventions to increase the physical activity of cardiac patients: a review. European Journal of Cardiovascular Prevention Rehabilitation, 18(1), 15-32.
8. Abdullah, M. R., Salim, S., and Jamil, M. M. A. (2013, November). Telemedicine system: Wireless healthcare units via RF, GSM, Bluetooth and PDA. In 2013 IEEE International Conference on Control System, Computing and Engineering (pp. 215-220). IEEE.
9. Bennett, J., Rokas, O., and Chen, L. (2017). Healthcare in the smart home: A study of past, present and future. Sustainability, 9(5), 840.
10. Telemadrid, (2015), Desarrollan una 'app' para la rehabilitación cardíaca domiciliaria. http://www.telemadrid.es/noticias/madrid/Desarrollan-app-rehabilitacion-cardiaca-domiciliaria-0-1659434062--20150223024815.html. Last access: 23/06/2019.

11. American Heart Association, (2017), HF Path, http://www.heart.org/en/health-topics/heart-failure/heart-failure-tools-resources/hf-path-heart-failure-self-management-tool. Last access: 23/06/2019
12. Heart Failure Society of America, (2017), Heart Failure Society of America Mobile Web App: Keeping Track of Your Health Just Got Easier! https://www.hfsa.org/patient/patient-tools/patient-app/, Last access: 23/06/2019
13. Salvador, C. H., Carrasco, M. P., De Mingo, M. G., Carrero, A. M., Montes, J. M., Martin, L. S., ... and Monteagudo, J. L. (2005). Airmed-cardio: a GSM and Internet services-based system for out-of-hospital follow-up of cardiac patients. IEEE Transactions on information technology in Biomedicine, 9(1), 73-85.
14. Waqialla, M., and Razzak, M. I. (2016). An Ontology-based framework aiming to support cardiac rehabilitation program. Procedia Computer Science, 96, 23-32.
15. Isern, D., and Moreno, A. (2016). A systematic literature review of agents applied in healthcare. Journal of medical systems, 40(2), 43.
16. Croatti, A., Montagna, S., Ricci, A., Gamberini, E., Albarello, V., and Agnoletti, V. (2018). BDI personal medical assistant agents: The case of trauma tracking and alerting. Artificial intelligence in medicine.
17. Isern, D., Moreno, A., Snchez, D., Hajnal, ., Pedone, G., and Varga, L. Z. (2011). Agent-based execution of personalised home care treatments. Applied Intelligence, 34(2), 155-180.
18. Koutkias, V. G., Chouvarda, I., and Maglaveras, N. (2003, September). Multi-agent system architecture for heart failure management in a home care environment. In Computers in Cardiology, 2003 (pp. 383-386). IEEE.
19. Arango, M., Bahler, L., Bates, P., Cochinwala, M., Cohrs, D., Fish, R., ... and Lee, K. C. (1993). The touring machine system. Communications of the ACM, 36(1), 69-77.
20. Mller, J. P., Pischel, M., and Thiel, M. (1994, August). Modeling reactive behaviour in vertically layered agent architectures. In International Workshop on Agent Theories, Architectures, and Languages (pp. 261-276). Springer, Berlin, Heidelberg.
21. K. Hindriks (2014), The shaping of the agent-oriented mindset.quot;, International Workshop on Engineering Multi-Agent Systems, pp. 1-14.
22. Mller, J.P., and Fischer, K.(2014), Application Impact of Multi-Agent Systems and Technologies: A Survey. In: Agent-Oriented Software Engineering: Reflections on Architectures, Methodologies, Languages, and Frameworks. pp. 27-53. Springer, Berlin, Heidelberg.
23. Annicchiarico, R., Garcia, U. C., and Urdiales, C. (Eds.). (2008). Agent technology and e-Health. Springer Science Business Media.
24. Borg G.A. (1982) Psychophysical bases of perceived exertion. Medicine and Science in Sports and Exercise. 14:377-381.
25. Roscoe, A. W. (2010). Understanding concurrent systems. Springer Science Business Media.
26. Krzywicki, D., Byrski, A., and Kisiel-Dorohinicki, M. (2014). Computing agents for decision support systems. Future Generation Computer Systems, 37, 390-400.
27. JAVA Agent DEvelopment Framework, http://jade.tilab.com/". Last access: 23/06/2019
28. Akka, https://akka.io/, Last access: 23/06/2019
29. Polar H10 Heart Rate Sensor, https://www.polar.com/en/products/accessories, Last access: 23/06/2019.
30. Bharti, R. HIVE: An Agent Based Modeling Framework, Thesis - San Jose State University, 2016.

31. Ferber, J., and Weiss, G. (1999). Multi-agent systems: an introduction to distributed artificial intelligence (Vol. 1). Reading: Addison-Wesley.